UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE
United States Patent and Trademark Office
Address: COMMISSIONER FOR PATENTS
P.O. Box 1450
Alexandria, Virginia 22313-1450
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/855,199	05/14/2001	Vijaya Raghavan	04899-044001	8175

959 7590 11/02/2006

LAHIVE & COCKFIELD, LLP
ONE POST OFFICE SQUARE
BOSTON, MA 02109-2127

EXAMINER	
ALHIJA, SAIF A	

ART UNIT	PAPER NUMBER
2128	

DATE MAILED: 11/02/2006

Please find below and/or attached an Office communication concerning this application or proceeding.

Office Action Summary	Application No.	Applicant(s)	
	09/855,199	RAGHAVAN ET AL.	
	Examiner	Art Unit	
	Saif A. Alhija	2128	

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

Period for Reply

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) OR THIRTY (30) DAYS, WHICHEVER IS LONGER, FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

Status

1)☒ Responsive to communication(s) filed on *06 August 2006*.

2a)☐ This action is **FINAL**. 2b)☒ This action is non-final.

3)☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

Disposition of Claims

4)☒ Claim(s) *1-18 and 20-78* is/are pending in the application.

4a) Of the above claim(s) _____ is/are withdrawn from consideration.

5)☐ Claim(s) _____ is/are allowed.

6)☒ Claim(s) *1-18 and 20-78* is/are rejected.

7)☐ Claim(s) _____ is/are objected to.

8)☐ Claim(s) _____ are subject to restriction and/or election requirement.

Application Papers

9)☐ The specification is objected to by the Examiner.

10)☒ The drawing(s) filed on *24 August 2001* is/are: a)☒ accepted or b)☐ objected to by the Examiner.

Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).

Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).

11)☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

Priority under 35 U.S.C. § 119

12)☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).

a)☐ All b)☐ Some * c)☐ None of:

1.☐ Certified copies of the priority documents have been received.

2.☐ Certified copies of the priority documents have been received in Application No. _____.

3.☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

* See the attached detailed Office action for a list of the certified copies not received.

Attachment(s)

1) ☒ Notice of References Cited (PTO-892)
2) ☐ Notice of Draftsperson's Patent Drawing Review (PTO-948)
3) ☐ Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08) Paper No(s)/Mail Date _____.
4) ☐ Interview Summary (PTO-413) Paper No(s)/Mail Date. _____.
5) ☐ Notice of Informal Patent Application (PTO-152)
6) ☐ Other: _____.

DETAILED ACTION

1. Claims 1-18, and 20-78 have been presented for examination.

Response to Amendement

2. A request for continued examination under 37 CFR 1.114, including the fee set forth in 37 CFR 1.17(e), was filed in this application after final rejection. Since this application is eligible for continued examination under 37 CFR 1.114, and the fee set forth in 37 CFR 1.17(e) has been timely paid, the finality of the previous Office action has been withdrawn pursuant to 37 CFR 1.114. Applicant's submission filed on 9 August 2006 has been entered.

i) Examiner thanks the Applicant for agreeing to the change of title. An amendment to that effect is still required.

ii) The Examiner withdraws the objection to the Specification following Applicants amendment.

iii) The Examiner withdraws the 112 2nd rejections of claims 1, 12, 24, 32, 60, 66, 70 and 74 following Applicants amendment.

iv) The Examiner notes that following Applicants amendment claims 1, 12, 24, 32, 34, 61, 66, 70, and 74 appear to be an intended use by including the phrase "to be used in." It is further noted that a recitation of the intended use of the claimed invention must result in a structural difference between the claimed invention and the prior art in order to patentably distinguish the claimed invention from the prior art. If the prior art structure is capable of performing the intended use, then it meets the claim. This applies to all other instances of the phrase.

v) Applicant argues that the reference does not disclose making a function call to a state diagram or making a graphical function call within the state machine model. Paragraph 168 of the reference discloses **"In one embodiment of the invention, the state diagram information may include information specifying executable code or source code, referred to below as program code,**

associated with one or more states. For example, when the user is using a state diagram editor to create a state diagram, when the user places a state in the state diagram, in one embodiment the user may further specify program code for the state."

vi) Applicant argues that the reference does not disclose both a graphical representation of a finite state machine and a graphical representation of a function. Paragraph 165 of the reference discloses **"FIG. 19 illustrates an example in which a graphical program has been programmatically generated based on the state diagram of FIG. 1. As described above, the GPG program may comprise, may be associated with, or may interface with a state diagram editor. FIG. 19 illustrates an exemplary graphical user interface of a state diagram editor, in which the state diagram of FIG. 1 is shown. FIG. 19 also illustrates a block diagram portion corresponding to the "Number is NOT Prime" state of the state diagram of FIG. 1. In this block diagram portion, the user has manually added graphical code to specify program instructions to execute when this state is active. In this case, the added code is operable to display a user interface message indicating that the chosen number is not prime."** The finite state machine is graphically represented as can be seen in Figure 19. In addition, in one embodiment and for example, the entire state machine can be seen as a function, or the transitions between the states, or the states themselves since a function is merely a set of code instructions.

vii) Applicant argues that the reference does not disclose user input defining the graphical function. The Abstract of the reference discloses, **"In one embodiment, the graphical program may be dynamically (programmatically) updated as the state diagram is being interactively constructed by the user."** Therefore since a function is merely a set of code instructions it can encompass the state diagram, its transitions, its states, or certain segments thereof.

viii) Applicant argues that the reference does not disclose user input defining a function flow diagram. As per Paragraph 9, the reference discloses **"The diagram may have one or more of data**

flow, control flow and/or execution flow representations. In response to the user constructing a diagram or graphical program using the block diagram editor, data structures may be automatically constructed which characterize an execution procedure which corresponds to the displayed procedure. The graphical program may be compiled or interpreted by a computer."

ix) Applicant argues that the reference does not disclose a component representing a function and referenced by a state or a transition to call the function within the state of transition. **See section v and vi above. In addition see Paragraph 9, which discloses "a block diagram editor."**

x) Applicant argues that the reference does not disclose an executable state diagram. **See section v and vi above.**

xi) Applicant argues that the reference does not disclose function invocations dealing with proximity and priority. As was stated in the previous office action a creator's priority order can allow for closest function definition to execute. For example, the default state call can be made proximally closest to the function call originator. In addition, it is unclear how this limitation renders the claims patentably distinct from the reference. At the very least the capabilities of the reference render it functionally equivalent.

xii) Applicant argues that the reference does not disclose textually referencing the graphically represented function within the model to cause an invocation of the graphically represented function during execution of the model. Figures 8, 10, 12, and 14 for example show the referenced graphical functions with textual references.

xiii) Please note the additional citations provided in the preceding sections are relevant to their corresponding claim limitations cited below.

Claim Rejections - 35 USC § 102

The following is a quotation of the appropriate paragraphs of 35 U.S.C. 102 that form the basis for the rejections under this section made in this Office action:

(e) the invention was described in (1) an application for patent, published under section 122(b), by another filed in the United States before the invention by the applicant for patent or (2) a patent granted on an application for patent by another filed in the United States before the invention by the applicant for patent, except that an international application filed under the treaty defined in section 351(a) shall have the effects for purposes of this subsection of an application filed in the United States only if the international application designated the United States and was published under Article 21(2) of such treaty in the English language.

3. **Claim(s) 1-18, 20-78** are rejected under 35 U.S.C. 102(e) as being clearly anticipated by **Kodosky et al. "System and Method for Programmatically Generating a Graphical Program in Response to a State Diagram" U.S. Patent Application Publication # 2002/0083413 A1.**

Regarding Claim 1:

Kodosky et al. discloses a computer-implemented method comprising:

providing a graphical user interface for defining at least one function to be used in a graphical representation of a finite state machine where the graphical representation is an executable model of the finite state machine. **(Page 14, Paragraph 165, Lines 1-5. Figure 19)**

representing the at least one function graphically; **(Page 14, Paragraph 165, Lines 1-5. Figure 19)**

calling the function that is represented graphically from within the finite state machine. **(Page 15, Paragraph 166, Lines 11-15)**

Regarding Claim 2:

Kodosky et al. discloses the method of claim 1 wherein defining the at least one function further comprises using a function block. **(Page 14, Paragraph 165, Lines 7-9. Figure 19)**

Regarding Claim 3:

Kodosky et al. discloses the method of claim 2 wherein defining the at least one function further comprises using a function prototype. **(Page 2, Paragraph 11, Lines 1-6)**

Regarding Claim 4:

Kodosky et al. discloses the method of claim 1 wherein the defining step further comprises using a function flow diagram. **(Page 1, Paragraph 9, Lines 7-9)**

Regarding Claim 5:

Kodosky et al. discloses the method of claim 1 wherein the function is represented graphically as a diagram comprising graphical elements. **(Figure 8)**

Regarding Claim 6:

Kodosky et al. discloses the method of claim 1 further comprising modifying the at least one function through graphical diagramming. **(Figure 8)**

Regarding Claim 7:

Kodosky et al. discloses a system comprising:

a computer comprising a graphical user interface, memory, storage, and at least one input device; **(Page 6, Paragraph 63, Lines 1-4)**

a computer program residing on computer readable media having instructions to cause the computer to: receive user input defining at least one graphical function; **(Page 6, Paragraph 63, Lines 1-8)**

receive user input to use the at least one graphical function in a simulation. **(Page 1, Paragraph 9, Lines 1-2)**

Regarding Claim 8:

Kodosky et al. discloses the system of claim 7 wherein the user input defining the at least one graphical function is entered into a function block. **(Page 1, Paragraph 9, Lines 1-2)**

Regarding Claim 9:

Kodosky et al. discloses the system of claim 7 wherein the user input defining the at least one graphical function includes a function prototype. **(Page 2, Paragraph 11, Lines 1-6)**

Regarding Claim 10:

Kodosky et al. discloses the system of claim 7 wherein the user input defining the at least one graphical function comprises a function flow diagram. **(Page 1, Paragraph 9, Lines 7-9)**

Regarding Claim 11:

Kodosky et al. discloses the system of claim 10 wherein the function flow diagram is comprised of graphical elements. **(Figure 8)**

Regarding Claim 12:

Kodosky et al. discloses a computer program product, stored in a computer readable medium, comprising instructions to cause a computer to:

receive user input defining at least one graphical function for use in a finite state machine; **(Page 6, Paragraph 63, Lines 1-8)**

use the at least one graphical function in a simulation of a system represented by the finite state machine. **(Page 1, Paragraph 9, Lines 1-2)**

Regarding Claim 13:

Kodosky et al. discloses the computer program product of claim 12 wherein the user input defining the at least one graphical function is entered into a function block. **(Page 1, Paragraph 9, Lines 1-2)**

Regarding Claim 14:

Kodosky et al. discloses the computer program product of claim 12 wherein the user input defining the at least one graphical function includes a function prototype. **(Page 2, Paragraph 11, Lines 1-6)**

Regarding Claim 15:

Kodosky et al. discloses the computer program product of claim 12 wherein the user input comprises a function flow diagram. **(Page 1, Paragraph 9, Lines 7-9)**

Regarding Claim 16:

Kodosky et al. discloses the computer program product of claim 15 wherein the function flow diagram is a comprised of graphical elements. **(Figure 8)**

Regarding Claim 17:

Kodosky et al. discloses a system for modeling finite state machines comprising:

a computer comprising a graphical user interface, memory, storage, and at least one input device; **(Page 6, Paragraph 63, Lines 1-4)**

means to receive user input to define at least one graphical function; **(Page 6, Paragraph 63, Lines 1-8)**

means to represent the graphical function as a state flow diagram; **(Page 2, Paragraph 16, Lines 4-9)**

means to use the graphical function in a simulation of at least one finite state machine. **(Page 15, Paragraph 166, Lines 13-20)**

Regarding Claim 18:

Kodosky et al. discloses the system of claim 17 wherein the user input to define the at least one graphical function is entered into a function block. **(Page 1, Paragraph 9, Lines 1-2)**

Regarding Claim 20:

Kodosky et al. discloses the system of claim 17 wherein the user input defining the at least one graphical function includes a function prototype. **(Page 2, Paragraph 11, Lines 1-6)**

Regarding Claim 21:

Kodosky et al. discloses the system of claim 17 wherein the user input comprises a function flow diagram. **(Page 1, Paragraph 9, Lines 7-9)**

Regarding Claim 22:

Kodosky et al. discloses the system of claim 21 wherein the function flow diagram is comprised of graphical elements. (Figure 8)

Regarding Claim 23:

Kodosky et al. discloses the system of claim 21 further comprising means for hiding the display of the function flow diagram based upon user input. **(Page 15, Paragraph 169, Lines 7-12. Linking the non-graphical code does not involve adding it to the graphical program therefore it is hidden in the graphical environment.)**

Regarding Claim 24:

Kodosky et al. discloses a method of operating a data processing system having a graphical user interface comprising:

creating a graphical representation of a finite state machine and a graphical representation of a function for use in the graphical representation of the finite state machine; and **(Page 1, Paragraph 9, Lines 9-14)**

simulating a system represented by the finite state machine wherein the graphical representation is an executable model of the system. **(Page 1, Paragraph 10, Lines 10-13)**

Regarding Claim 25:

Kodosky et al. discloses the method of claim 24 wherein the graphical representation of the function comprises a function prototype. **(Page 2, Paragraph 11, Lines 1-6)**

Regarding Claim 26:

Kodosky et al. discloses the method of claim 25 wherein the function prototype defines a textual format for invoking the function. **(Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 27:

Kodosky et al. discloses the method of claim 26 wherein the graphical representation of the finite state machine includes at least one invocation of the function using the defined textual format. **(Page 12, Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 28:

Kodosky et al. discloses the method of claim 24 further comprising shadowing a function, wherein shadowing comprising using in a function invocation a function definition closest to a point of invocation of the function in a state diagram hierarchy. **(Page 3, Paragraph 20, Lines 8-13; Creators priority order can allow for closest function definition to execute.)**

Regarding Claim 29

Kodosky et al. discloses the method of claim 24 wherein the function is exportable by a state chart and may be invoked anywhere in the finite state machine in which the chart appears, including other charts that define the finite state machine. **(Page 3, Paragraph 26, Lines 4-10. Page 9, Paragraph 100, Lines 1-5)**

Regarding Claim 30:

Kodosky et al. discloses the method of claim 24 wherein simulating the system represented by the finite state machine further comprises computer code generation. **(Page 12, Paragraph 133, Lines 1-4)**

Regarding Claim 31:

Kodosky et al. discloses the method of claim 24

wherein the graphical representation of the function comprises a function prototype defining a textual format for invoking the function; **(Page 12, Paragraph 132, Lines 1-5. Figure 8)**

and wherein the graphical representation of the finite state machine includes an invocation of the function using the defined textual format. **(Page 12, Paragraph 132, Lines 1-5. Figure 8)**

Regarding Claim 32:

Kodosky et al. discloses a computer readable medium having encoded thereon

instructions for causing a computer system to receive through a graphical user interface graphical representation of a finite state machine and a graphical representation of at least one function for use in the graphical representation of the finite state machine; and **(Page 1, Paragraph 9, Lines 9-14)**

instructions for simulating a system represented by the finite state machine where the graphical representation is an executable model of the system. **(Page 1, Paragraph 10, Lines 10-13)**

Regarding Claim 33:

Kodosky et al. discloses the computer readable medium of claim 32,

wherein the graphical representation of the function comprises a function prototype defining a textual format for invoking the function; **(Page 2, Paragraph 11, Lines 1-6)**

and wherein the graphical representation of the finite state machine includes an invocation of the function using the define textual format. **(Page 12, Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 34:

Kodosky et al. discloses in an electronic device, a method of graphically representing an event-driven system, comprising:

Providing one or more block components representing one or more states in an executable model; **(Page 1, Paragraph 9, Lines 1-3)**

Providing one or more transition components representing transitions between the one or more block states; **(Page 2, Paragraph 16, Lines 1-4)** and

Providing a component representing a function referenced by at least one the states or at least one of the transitions to call the function at the at least one of the states or the at least one of the transitions. **(Page 2, Paragraph 17, Lines 1-4)**

Regarding Claim 35:

Kodosky et al. discloses the method of claim 34, wherein the function accepts at least one argument and returns at least one result. **(Page 1, Paragraph 9, Lines 1-4)**

Regarding Claim 36:

Kodosky et al. discloses the method of claim 34, further comprising invoking the function at one or more transition components. **(Page 12, Paragraph 132 Lines 1-5. Figure 8. Page 1, Paragraph 10, Lines 1-5)**

Regarding Claim 37:

Kodosky et al. discloses the method of claim 34 further comprising specifying data properties of the function. **(Page 1, Paragraph 9, Lines 7-9)**

Regarding Claim 38:

Kodosky et al. discloses the method of claim 34 further comprising associating a data item with the function. **(Page 1, Paragraph 9, Lines 7-9. Page 2, Paragraph 11, Lines 2-7)**

Regarding Claim 39:

Kodosky et al. discloses the method of claim 34, wherein the function comprises a graphical function. **(Page 6, Paragraph 63, Lines 1-8)**

Regarding Claim 40:

Kodosky et al. discloses the method of claim 34, wherein the function has a plurality of configurable properties. **(Page 1, Paragraph 10, Lines 1-5)**

Regarding Claim 41:

Kodosky et al. discloses the method of claim 34, wherein the function defines a textual format for invoking the function. **(Page 12, Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 42:

Kodosky et al. discloses the method of claim 34, further comprising providing a shadowing function, wherein shadowing comprises using in a function invocation a function definition proximally closest to a point of invocation of the function in a state diagram hierarchy. **(Page 3, Paragraph 20, Lines 8-13; Creators priority order can allow for closest function definition to execute.)**

Regarding Claim 43:

Kodosky et al. discloses in a graphical representation environment, a system for graphically representing an event-driven system, comprising:

One or more bock components representing one or more states in an executable model; **(Page 1, Paragraph 9, Lines 1-3)**

One or more transition components representing transitions between the one or more block components representing the states; **(Page 2, Paragraph 16, Lines 1-4) and**

A component representing a function and referenced by at least one of the states or at least one of the transitions to call the function at one of the states or one of the transition. **(Page 2, Paragraph 17, Lines 1-4)**

Regarding Claim 44:

Kodosky et al. discloses the system of claim 43, wherein the function accepts at least one argument and returns at least one result. **(Page 1, Paragraph 9, Lines 1-4)**

Regarding Claim 45:

Kodosky et al. discloses the system of claim 43, wherein at least a subset of the one or more block components representing the states and the one or more transition components can invoke the function. **(Page 12, Paragraph 132 Lines 1-5. Figure 8. Page 1, Paragraph 10, Lines 1-5)**

Regarding Claim 46:

Kodosky et al. discloses the system of claim 43, further comprising means for specifying data properties of the function. **(Page 1, Paragraph 9, Lines 7-9)**

Regarding Claim 47:

Kodosky et al. discloses the system of claim 43, further comprising means for associating a data item with the function. **(Page 1, Paragraph 9, Lines 7-9. Page 2, Paragraph 11, Lines 2-7)**

Regarding Claim 48:

Kodosky et al. discloses the system of claim 34, wherein the function comprises a graphical function. **(Page 6, Paragraph 63, Lines 1-8)**

Regarding Claim 49:

Kodosky et al. discloses the system of claim 43, wherein the function has a plurality of configurable properties. **(Page 1, Paragraph 10, Lines 1-5)**

Regarding Claim 50:

Kodosky et al. discloses the system of claim 43, wherein the function defines a textual format for invoking the function. **(Page 12, Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 51:

Kodosky et al. discloses the system of claim 43, further comprising means for providing a shadowing function, wherein shadowing comprises using in a function invocation a function definition proximally closest to a point of invocation of the function in a state diagram hierarchy. **(Page 3, Paragraph 20, Lines 8-13; Creators priority order can allow for closest function definition to execute.)**

Regarding Claim 52:

Kodosky et al. discloses a medium for use in a graphical representation environment on an electronic device, the medium holding instructions executable using the electronic device for graphically representing an event-driven system, said instructions comprising instructions of:

Providing one or more block components representing one or more states in an executable model; **(Page 1, Paragraph 9, Lines 1-3)**

Providing one or more transition components representing transitions between the one or more block components representing the states; **(Page 2, Paragraph 16, Lines 1-4)** and

Providing a block component representing a function and reference by at least one of the states or at least one of the transitions to call the function at one of the states or one of the transitions. **(Page 2, Paragraph 17, Lines 1-4)**

Regarding Claim 53:

Kodosky et al. discloses the medium of claim 52, wherein the function accepts at least one argument and returns at least one result. **(Page 1, Paragraph 9, Lines 1-4)**

Regarding Claim 54:

Kodosky et al. discloses the medium of claim 52, wherein the one or more transition components can invoke the function. **(Paragraph 132 Lines 1-5. Figure 8. Page 1, Paragraph 10, Lines 1-5)**

Regarding Claim 55:

Kodosky et al. discloses the medium of claim 52, further comprising instructions for accepting user input specifying data properties of the function. **(Page 1, Paragraph 9, Lines 7-9)**

Regarding Claim 56:

Kodosky et al. discloses the medium of claim 52, further comprising instructions for associating a data item with the function. **(Page 1, Paragraph 9, Lines 7-9. Page 2, Paragraph 11, Lines 2-7)**

Regarding Claim 57:

Kodosky et al. discloses the medium of claim 52, wherein the function comprises a graphical function. **(Page 6, Paragraph 63, Lines 1-8)**

Regarding Claim 58:

Kodosky et al. discloses the medium of claim 52, wherein the function has a plurality of configurable properties. **(Page 1, Paragraph 10, Lines 1-5)**

Regarding Claim 59:

Kodosky et al. discloses the medium of claim 52, wherein the function defines a textual format for invoking the function. **(Page 12, Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 60:

Kodosky et al. discloses the medium of claim 52, further comprising instructions for providing a shadowing function wherein shadowing comprises using in a function invocation a function definition proximally closest to a point of invocation of the function in a state diagram hierarchy. **(Page 3, Paragraph 20, Lines 8-13; Creators priority order can allow for closest function definition to execute.)**

Regarding Claim 61:

Kodosky et al. discloses A computer implemented method for modeling a system using a graphical block diagram environment, said method comprising:

graphically representing a function for use in an executable model within the graphical block diagram environment; **(Page 14, Paragraph 165, Lines 1-5. Figure 19)** and

textually referencing the graphically represented function within the model to cause an invocation of the graphically represented function during execution of the model. **(Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 62:

Kodosky et al. discloses The computer implemented method of claim 61, wherein the model is represented as a finite state machine. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 63:

Kodosky et al. discloses The computer implemented method of claim 62, wherein the finite state machine is a hierarchical finite state machine. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 64:

Kodosky et al. discloses The computer implemented method of claim 62 further comprising:

Associating the graphically represented function with at least one state or transition within the finite state machine. **(Page 2, Paragraph 16, Lines 1-4)**

Regarding Claim 65:

Kodosky et al. discloses The computer implemented method of claim 61, wherein the graphically represented function is represented as at least one of a finite state machine, a state flow diagram, a function flow diagram, and a graphical block diagram model. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 66:

Kodosky et al. discloses A medium holding instructions executable using the electronic device for modeling a system using a graphical block diagram environment, said instructions comprising instructions for:

Graphically representing a function for use in an executable model within the graphical block diagram environment; **(Page 14, Paragraph 165, Lines 1-5. Figure 19)**

textually referencing the graphically represented function within the model to cause an invocation of the graphically represented function during execution of the model. **(Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 67:

Kodosky et al. discloses The medium of claim 66, wherein the model is represented as a finite state machine. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 68:

Kodosky et al. discloses The medium of claim 67 further comprising instructions for:

Associating the graphically represented function with at least one state of transition within the finite state machine. **(Page 2, Paragraph 16, Lines 1-4)**

Regarding Claim 69:

Kodosky et al. discloses The medium of claim 66, wherein the graphically represented function is represented as at least one or a combination of:

a finite state machine, **(Page 3, Paragraph 20, Lines 8-13)**

a state flow diagram,

a function flow diagram,

and a graphical block diagram model.

Regarding Claim 70:

Kodosky et al. discloses A computer implemented system for modeling using a graphical block diagram environment, said system comprising:

Means for graphically representing a function for use in an executable model within the graphical block diagram environment; **(Page 14, Paragraph 165, Lines 1-5. Figure 19)** and

Means for textually referencing the graphically represented function within the model to cause an invocation of the graphically represented function during execution of the model. **(Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 71:

Kodosky et al. discloses The system of claim 70, wherein the model is represented as a finite state machine. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 72:

Kodosky et al. discloses The system of claim 71 further comprising means for associating the graphically represented function with at least one state of transition within the finite state machine. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 73:

Kodosky et al. discloses The system of claim 70, wherein the graphically represented function is represented as at least one or a combination of a finite state machine, a state flow diagram, a function flow diagram, and a graphical block diagram model. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 74:

Kodosky et al. discloses A graphical block diagram modeling system comprising:

A graphical function for use in an executable model; **(Page 14, Paragraph 165, Lines 1-5. Figure 19)** and

A graphical representation of the model including a textual reference of the graphically represented function within the graphical representation of the model to cause an invocation of the graphical function during execution of the model. **(Paragraph 132 Lines 1-5. Figure 8)**

Regarding Claim 75:

Kodosky et al. discloses The system of claim 74, wherein the model is represented as a finite state machine. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 76:

Kodosky et al. discloses The system of claim 75, wherein the finite state machine is a hierarchical finite state machine. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 77:

Kodosky et al. discloses The system of claim 75, wherein the finite state machine further comprises at least one state or transition associated with the graphical function. **(Page 3, Paragraph 20, Lines 8-13)**

Regarding Claim 78:

Kodosky et al. discloses The system of claim 74, wherein the graphical function is represented as at least one or a combination of:

a finite state machine, **(Page 3, Paragraph 20, Lines 8-13)**

a state flow diagram,

a function flow diagram,

and a graphical block diagram model.

Conclusion

4. The prior art made of record is not relied upon because it is cumulative to the applied rejection. These references include:

A) Stateflow Version 2. Mathworks. May 1999.

5. All Claims are rejected.

6. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Saif A. Alhija whose telephone number is (571) 272-8635. The examiner can normally be reached on M-F, 11:00-7:30.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Kamini Shah can be reached on (571) 272-22792279. The fax phone number for the organization where this application or proceeding is assigned is (571) 273-8300.

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see http://pair-direct.uspto.gov. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

SAA

October 22, 2006

HUGH JONES Ph.D.
PRIMARY PATENT EXAMINER
TECHNOLOGY CENTER 2100